PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application Of: Layzell, et al.

Serial No.: 10/652,787

Filed: 08/29/2003

For: **Page Composition**

Group No.: 2673

Docket No. 200208258-2

Confirmation No.: 3400

CLAIM OF PRIORITY TO AND SUBMISSION OF CERTIFIED COPY OF UNITED KINGDOM APPLICATION PURSUANT TO 35 U.S.C. §119

Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

In regard to the above-identified pending patent application and in accordance with 35 U.S.C. §119, Applicant hereby claims priority to and the benefit of the filing date of United Kingdom patent application entitled, "Page Composition", filed August 30, 2002, and assigned serial number 0220166.3. Further pursuant to 35 U.S.C. §119, enclosed is a certified copy of the United Kingdom patent application

Respectfully Submitted,

THOMAS, KAYDEN, HORSTEMEYER & RISLEY, L.L.P.

By: Scott A. Horstemeyer

Scott A. Horstemeyer, Reg. No. 34,183

100 Galleria Parkway, Suite 1750
Atlanta, Georgia 30339
770-933-9500

CERTIFIED MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as **"First Class Mail,"** in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on May 10, 2004

Julie Campbell
Signature

20020825 8-2

The Patent Office
PATENTS • DESIGNS • COPYRIGHT • TRADE MARKS

INVESTOR IN PEOPLE

The Patent Office
Concept House
Cardiff Road
Newport
South Wales
NP10 8QQ

I, the undersigned, being an officer duly authorised in accordance with Section 74(1) and (4) of the Deregulation & Contracting Out Act 1994, to sign and issue certificates on behalf of the Comptroller-General, hereby certify that annexed hereto is a true copy of the documents as originally filed in connection with the patent application identified therein.

In accordance with the Patents (Companies Re-registration) Rules 1982, if a company named in this certificate and any accompanying documents has re-registered under the Companies Act 1980 with the same name as that with which it was registered immediately before re-registration save for the substitution as, or inclusion as, the last part of the name of the words "public limited company" or their equivalents in Welsh, references to the name of the company in this certificate and any accompanying documents shall be treated as references to the name with which it is so re-registered.

In accordance with the rules, the words "public limited company" may be replaced by p.l.c., plc, P.L.C. or PLC.

Re-registration under the Companies Act does not constitute a new legal entity but merely subjects the company to certain additional company law rules.

Signed M C...

Dated 3 September 2003

Patents Form 1/77

Patents Act 1977
(Rule 16)

THE PATENT OFFICE
A
30 AUG 2002
RECEIVED BY FAX

The Patent Office

30AUG02 E744785-1 D01463
P01/7700 0.00-0220166.3

Request for grant of a patent

(See the notes on the back of this form. You can also get an explanatory leaflet from the Patent Office to help you fill in this form)

The Patent Office

Cardiff Road
Newport
South Wales
NP10 8QQ

1. Your reference — 200208258-1 GB

2. Patent application number *(The Patent Office will fill in this part)* — 0220166.3 — 30 AUG 2002

3. Full name, address and postcode of the or of each applicant *(underline all surnames)*

   Hewlett-Packard Company
   3000 Hanover Street
   Palo Alto
   CA 94304, USA

   Patents ADP number *(if you know it)* — 06293385001

   If the applicant is a corporate body, give the country/state of its incorporation — Delaware, USA

4. Title of the invention — Page Composition

5. Name of your agent *(if you have one)*

   "Address for service" in the United Kingdom to which all correspondence should be sent *(including the postcode)*

   Richard A. Lawrence
   Hewlett-Packard Ltd, IP Section
   Filton Road, Stoke Gifford
   Bristol BS34 8QZ

   Patents ADP number *(if you know it)* — 07448038001

6. If you are declaring priority from one or more earlier patent applications, give the country and the date of filing of the or of each of these earlier applications and *(if you know it)* the or each application number

Country	Priority application number *(if you know it)*	Date of filing *(day / month / year)*

7. If this application is divided or otherwise derived from an earlier UK application, give the number and the filing date of the earlier application

Number of earlier application	Date of filing *(day / month / year)*

8. Is a statement of inventorship and of right to grant of a patent required in support of this request? *(Answer 'Yes' if:*
   a) any applicant named in part 3 is not an inventor, or
   b) there is an inventor who is not named as an applicant, or
   c) any named applicant is a corporate body.
   See note (d))

   Yes

Patents Form 1/77

9. Enter the number of sheets for any of the following items you are filing with this form. Do not count copies of the same document

Continuation sheets of this form	
Description	17
Claim(s)	5
Abstract	1
Drawing(s)	10

10. If you are also filing any of the following, state how many against each item.

Priority documents	-
Translations of priority documents	-
Statement of inventorship and right to grant of a patent *(Patents Form 7/77)*	1
Request for preliminary examination and search *(Patents Form 9/77)*	1
Request for substantive examination *(Patents Form 10/77)*	-
Any other documents *(please specify)*	Fee Sheet

11. I/We request the grant of a patent on the basis of this application.

Signature Richard A. Lawrence Date 30/8/02

12. Name and daytime telephone number of person to contact in the United Kingdom — Meg Joyce Tel: 0117-312-9068

Warning

After an application for a patent has been filed, the Comptroller of the Patent Office will consider whether publication or communication of the invention should be prohibited or restricted under Section 22 of the Patents Act 1977. You will be informed if it is necessary to prohibit or restrict your invention in this way. Furthermore, if you live in the United Kingdom, Section 23 of the Patents Act 1977 stops you from applying for a patent abroad without first getting written permission from the Patent Office unless an application has been filed at least 6 weeks beforehand in the United Kingdom for a patent for the same invention and either no direction prohibiting publication or communication has been given, or any such direction has been revoked.

Notes

a) *If you need help to fill in this form or you have any questions, please contact the Patent Office on 08459 500505.*

b) *Write your answers in capital letters using black ink or you may type them.*

c) *If there is not enough space for all the relevant details on any part of this form, please continue on a separate sheet of paper and write "see continuation sheet" in the relevant part(s). Any continuation sheet should be attached to this form.*

d) *If you have answered 'Yes' Patents Form 7/77 will need to be filed.*

e) *Once you have filled in the form you must remember to sign and date it.*

f) *For details of the fee and ways to pay please contact the Patent Office.*

0045747 30-Aug-02 04:39

PDNO 200208258 DUPLICATE

PAGE COMPOSITION

Field of the Invention

This invention relates to page composition for documents, and is particularly relevant to generation of custom documents from a plurality of objects.

Description of Prior Art

Composition of document pages is a difficult task to achieve in a practical and aesthetically satisfactory manner. It is generally an integral part of the composition of a new document. However, it is frequently the case that a new document is to be created from, in whole or in part, a collection of pre-existing objects. This task will normally be taken by a human professional with graphic design skills. Solutions for automating this task are limited. The conventional solution is to use a template for a page, and to fit the chosen pre-existing objects into the space on a template (this approach is used, for example, in the personalised recommendation templates provided by websites such as www.amazon.com). Recently it has been suggested by Lisa Purvis in "A Genetic Algorithm Approach to Automated Custom Document Assembly", Proceedings of the Second International Workshop on Intelligent Systems Design and Applications (ISDA 2002), August 2002, Atlanta, USA that custom document assembly from existing parts using a genetic algorithm may satisfy content and layout constraints and fulfil certain desired design properties by specification of a group of high-order constraints. Such an approach requires limited user interaction but is computationally complex.

Summary of the Invention

Accordingly, in a first aspect of the invention there is provided a method of composing a page, or a portion of a page, of a document, comprising: defining a plurality of objects to be fitted on to the page and dimensional attributes of each of the objects; establishing an arrangement of the plurality of objects such that each object lies within a separate rectangle of a slicing structure dissection of a rectangular area; establishing a function which provides a total cost of an arrangement of the plurality of objects based on one or more properties of the arrangement; and finding a slicing

structure arrangement of the plurality of objects with a minimised total cost by means of an iterative process.

It should be noted that in this context, dimensional attributes of an object need not be a height and width coordinate, but may be any aspect of the dimensions of the object necessary to allow an arrangement to be determined and a cost evaluated. For example, dimensional attributes may be an area or range of areas and an aspect ratio or range of aspect ratios.

In a second aspect of the invention there is provided a method of composing a page, or a portion of a page, of a document, comprising: defining a plurality of objects to be fitted on to the page and dimensional attributes of each of the objects; establishing a function to represent a total area of an arrangement of the plurality of objects; minimising the function to find a minimised total area arrangement; and fitting the minimised total area arrangement to the page.

In a third aspect of the invention there is provided a method of providing a customised document having a plurality of pages, comprising: selecting a plurality of selected objects for inclusion in the document from a database of two-dimensional objects, and assigning each of the selected objects to one of a plurality of groups; assigning each of the selected objects to one of the pages of the document; arranging the objects assigned to each one of the pages in an arrangement such as to minimise a function dependent on a total area of the arrangement and on proximity to each other of objects in the same group.

A processor of a computer system may be programmed to carry out a method according to any of these aspects. Such programming may be achieved by use of a signal or data carrier having code adapted to program the processor accordingly.

Brief Description of the Drawings

Specific embodiments of the invention will now be described, by way of example, with reference to the accompanying drawings, of which:

Figure 1 shows an arrangement of objects on a page of a document;

Figure 2 shows the dissection of a rectangle into a slicing structure as employed in embodiments of the invention;

Figure 3 shows the slicing structure of Figure 2 represented as a slicing tree;

Figure 4 shows the slicing structure of Figure 2 represented as a Polish expression;

Figure 5 illustrates a first mutation operation used in a genetic algorithm operating on the slicing structure of Figure 2 according to one embodiment of the invention;

Figure 6 illustrates a second mutation operation used in a genetic algorithm operating on the slicing structure of Figure 2 according to one embodiment of the invention;

Figure 7 illustrates a third mutation operation used in a genetic algorithm operating on the slicing structure of Figure 2 according to one embodiment of the invention;

Figure 8 illustrates a first crossover operation used in a genetic algorithm operating on the slicing structure of Figure 2 according to one embodiment of the invention;

Figure 9 illustrates a second crossover operation used in a genetic algorithm operating on the slicing structure of Figure 2 according to one embodiment of the invention;

Figure 10 illustrates a third crossover operation used in a genetic algorithm operating on the slicing structure of Figure 2 according to one embodiment of the invention;

Figure 11 shows the third crossover operation of Figure 10 in relation to a tree structure;

Figures 12A, 12B and 12C show bounding curves indicating the dimensions of rectangles into which objects of constant area and variable aspect ratio can be fitted;

Figure 13 illustrates how the bounding curve of an expression can be derived from the bounding curves of the two operands;

Figure 14 shows an example of a composed page according to an embodiment of the invention;

Figures 15A, 15B and 15C show examples of composed pages with the same set of two groups of objects composed according to an embodiment of the invention with different relative cost weightings to total area and to separation of objects in a group;

Figure 16 represents steps involved in a process of producing a customised document to which aspects of the present invention are applicable; and

Figure 17 shows a computing system suitable for carrying out embodiments of the invention and for consuming the results thereof.

Detailed Description of Specific Embodiments of the Invention

A method of providing a customised document according to embodiments of the invention will now be described.

Basic steps of a document production process are shown in Figure 16. The initial step 161 is to determine what content the document needs to contain. The document may be, for example, a brochure tailored to the interests of the intended recipient – in this example, we shall assume the case of a holiday brochure. For this stage of the process, any of a number of conventional approaches could be used both to determine the interests of the intended recipient and to make a selection of content items. One such conventional approach is outlined as follows. The content items are a collection of viewable or printable two-dimensional elements, all relating to holidays: these may be pictures of locations, text descriptions of holiday packages, text descriptions of flights and so on. Each is tagged with one or more descriptors indicating their relevance to a particular keyword. The significance of the keywords for the intended recipient is determined by direct polling of the recipient, by analysing past holiday choices made by the recipient, or by studying web pages viewed by the recipient or by some combination of some or all of these. The significance of the keywords to the intended recipient is combined with the relevance of the keywords to the content items to provide a selection score for each content item, and the content items above a

threshold value are selected for inclusion. This is merely one exemplary approach among many, and it should be noted that essentially any approach for determining particularly relevant content items in a database (such as, for example, approaches used for selection of content by search engines operating on the World Wide Web) can be used in connection with aspects of the present invention.

For particular aspects of the invention, it is appropriate that the selected content items are divided into a number of groups 162. This again can be achieved in a number of ways: for example, a content item may be assigned to a group on its entry into the database, or may be assigned after selection to a group determined by a keyword to which it is most relevant.

Once selected, for a multiple page document it will be necessary to assign 163 selected content items to a page. This again can be achieved in a number of ways (according to a predetermined order of groups, in accordance with greatest interest scores for the intended recipient, or otherwise) and need not in all cases be an irrevocable assignment (it may be affected by subsequent inability to produce a satisfactory arrangement of content items, for example). Again, aspects of the present invention can be employed in accordance with essentially any strategy for allocating content items to pages. The number of pages in the document may also be determined in accordance with the number, or total size, of data items to be provided (of course, selection criteria may also be tightened or relaxed so that the amount of content matches the space available).

The next step is that of primary interest in application of aspects of the present invention – the arrangement 164 of selected content items allocated to a document page on that document page. This will be discussed in much greater detail below. There may be additional document reorganisation steps after that point – particularly if it has not been possible to produce a satisfactory arrangement for the or any one of the pages, in which case it may be necessary to transfer content items from page to page, or to add further pages to the document – but the only remaining step to be generally expected is matching 165 of the arrangement to the viewable region of the page. This may involve a scaling or expansion process – again, this will be described in greater detail below.

This process can be achieved on conventional computational hardware programmed with appropriate software (provided according to aspects of the invention). An appropriate system is shown in Figure 17. The steps of Figure 16 may be carried out by an appropriately programmed processor 171 with access to a memory 172, for example here in server 173. The result is to be rendered on display 174 of a client computer 175, or, in cases of particular interest here, to be printed on printer 176 (which may be of essentially any type – a laser printer is shown here, though for the case of custom publishing a preferred solution may be a high performance digital printer such as the HP Indigo Press w3200).

The result for a document page may be such as that shown in Figure 1. Two content items in a group relating to "Frogs" – picture 11 and text block 12 – lie together at the top of a page 10, whereas three content items in a group relating to "Lions" – picture 13, text block 14 and mixed block 15 – lie together at the bottom of the page 10. Note that the border 16 of the page 10 is a visually apparent border, whereas the border 17 of a content item is not (necessarily) a visible border, but may have significance only in the process of page composition. There will now be described an approach by which a logical and visually satisfactory grouping of content elements may be achieved in accordance with aspects of the present invention.

It can be seen that content items here are, visually, two-dimensional objects that may be fitted on to a page of a document. In certain aspects of the invention, it is appropriate to represent content items as rectangular with the same axes as for the page (if the informative content is not rectangular, the content item may have the dimensions of, for example, the smallest such rectangle that could bound the informative content).

The present inventors have appreciated that particular advantages can be gained from representing content items as rectangular objects with the same axes as for the page. In particular, they have appreciated that considerable computational advantages can be gained while still achieving very effective results. A significant advantage is that it becomes possible to use the mathematics of rectangle dissection – rectangle dissection can be defined as subdivision of a given rectangle by horizontal and vertical line

segments into a finite number of non-overlapping rectangles. Cutting a rectangle can be defined as dividing the rectangle into two rectangles by a horizontal or a vertical line. Particular aspects of the invention involve the use of a slicing structure – a slicing structure may be defined as a rectangle dissection that can be obtained by recursively cutting rectangles into smaller rectangles. An example is shown in Figure 2. Rectangle 20 is divided by a first, horizontal, cut 21 into a rectangle 5 and a remainder rectangle. The remainder rectangle is divided by a second, vertical, cut 22 into a rectangle 4 and a second remainder rectangle. The second remainder rectangle is then divided by a third, horizontal, cut 23 into a rectangle 3 and a third remainder rectangle. Finally, this third remainder rectangle is divided by a fourth, vertical, cut 24 into two rectangles 1, 2.

It will be appreciated that a slicing structure can readily be depicted as a binary tree. Such a tree, known as a slicing tree, is shown in Figure 3. This shows a representation of the slicing structure of Figure 2, with horizontal cuts 21 and 23 now represented by a horizontal cut operator + and vertical cuts 22 and 24 now represented by a vertical cut operator *.

Use of slicing structures to optimise VLSI circuit layout is discussed in D.F.Wong and C.L.Liu, "A New Algorithm for Floorplan Design", Proc. 23rd ACM/IEEE Design Automation Conference, Las Vegas, NV, 1986, 101-107, the contents of which are incorporated by reference herein. Wong and Liu developed a Polish notation for representing slicing structures. Polish notations list operands of functions before (strictly this is reverse Polish) or after their operator – this enables a sequence of operands and operators to be built up which does not require the use of brackets. Figure 4 shows the slicing tree of Figure 3 rendered in this Polish notation – the first "root" cut 21 is found at the end of the expression, preceded by the subtree rooted at cut 22 as the first operand and rectangle 5 as the second operand. The remainder of the expression can be seen to describe the rest of the tree according to the same principles. Wong and Liu found that a normalized Polish expression for a slicing structure (in their normalized expressions there are no consecutive operators of the same type) provided a unique representation of a slicing structure. Wong and Liu attempted to solve the VLSI circuit layout problem by using simulated annealing (a well-known mathematical technique for solving complex state problems by "cooling"

a system – when "warm", the system can make changes that have an associated "energy" cost, in the same way that a physical system can change at elevated temperature, but on progressive "cooling" fewer changes can be made until a solution is "frozen" – this approach is effective as systems migrate towards "energy" minima). Wong and Liu allowed three types of change to be made (set out here as Change 1, Change 2 and Change 3) and required the result to be a normalized Polish expression.

Wong and Liu's work was built upon in J.P.Cohoon, S.U.Hegde, W.N.Martin and D.Richards, "Floorplan Design Using Distributed Genetic Algorithms", IEEE International Conference on Computer Aided-Design 1988, November 198,. IEEE, New York, 452-455, the contents of which are incorporated by reference herein. Cohoon et al developed Wong and Liu's approach by using a genetic algorithm rather than simulated annealing. Unlike simulated annealing (which starts with one candidate expression and then makes a series of allowed changes to it), a genetic algorithm operates on a population of candidate expressions by producing small variations, the results of which ("offspring") are given a "fitness" score relating to their effectiveness as a solution and which affects the likelihood of their being involved in production of the next generation of candidate expressions. Cohoon et al used Wong and Liu's changes – as these involve only one expression, these are considered "mutations" in respect of a genetic algorithm – and also further changes (set out here as Change 4, Change 5 and Change 6) which involve a "crossover" between two parent expressions. In Cohoon et al's approach, any Polish expression able to represent a slicing structure may be used, and not merely normalized Polish expressions.

In a preferred embodiment of the present invention, a genetic algorithm is used to find minimised values of a function. Preferred functions will be discussed further below, but the mechanics of the genetic algorithm (which can be used with a multitude of functions) will be discussed first. The different changes that can be used to create offering will now be discussed with reference to Figures 5 to 11.

<u>Change 1</u> - This is shown in Figure 5. It is a mutation from a single expression 50, and involves the transposition of two adjacent operands 53, 54. If the

initial expression 50 is a normalized Polish expression, the mutated expression 51 will also be a normalized Polish expression.

Change 2 - This is shown in Figure 6. This is the second mutation change from a single expression 60, and involves taking the complement 64 of a chain 63 of operators (a sequence of operators uninterrupted by operands), wherein to complement a chain involves transforming every + to a * and every * to a +. Again, if the initial expression 60 is a normalized Polish expression, the mutated expression 61 will also be a normalized Polish expression.

Change 3 - This is shown in Figure 7. This is the third mutation change, and involves transposing an adjacent operator 72 and operand 73 in the initial expression 70 to form the mutated expression 71. Unlike Change 1 and Change 2, Change 3 does not necessarily produce a normalized Polish expression – in fact, the mutated expression may not describe a possible slicing structure. The results of Change 3 will therefore need to be checked to ensure that they do describe a slicing structure.

Change 4 - This is shown in Figure 8. This is the first crossover change from two parent expressions, and involves copying the operands from first parent 80 into identical positions in the offspring 82, and then to add operators into the gaps in the same sequence as which they occur in second parent 81. The action of the change is to propagate groups of operands from the first parent to the next generation. The result is a well-formed Polish expression (ie it does describe a slicing structure) but not necessarily a normalized Polish expression.

Change 5 - This is shown in Figure 9. This is the second crossover change from two parent expressions, and involves copying the operators from first parent 90 into identical positions in the offspring 92, and then to add operands into the gaps in the same sequence as which they occur in second parent 91. The action of the change is to propagate the slicing of the first parent to the next generation. Again, the result is a well-formed Polish expression but not necessarily a normalized Polish expression.

<u>Change 6</u> - This is shown in Figure 10. This is the third crossover change from two parent expressions, and is more complex than Changes 4 and 5. Initially, the first parent 100 is copied, and then an operator 103 is selected at random. The operands 104 of this operator 103 are preserved, but all other operands in the offspring 102 are then re-ordered using the order in which they appear in the second parent 101. This can be seen more clearly from the slicing tree 110 shown in Figure 11. A complete subtree 112 is retained from the slicing tree 110 whereas the slicing of the other structure 111 is retained although the operands may be changed. Again, the result is a well-formed Polish expression but not necessarily a normalized Polish expression.

The present inventors have appreciated that the solution approaches developed by Wong and Liu and by Cohoon et al may be very effectively applied to the apparently dissimilar problem of page decomposition. Prime requirements in VLSI floorplan layout are minimisation of total area occupied by components and minimisation of wire length between components. Wong and Liu and Cohoon are both attempting to find floorplan solutions which meet these goals together, and do this by determining a cost function relevant to both objectives which determines whether a given solution is best (or fittest). In Wong and Liu and Cohoon, the components occupy the rectangles of the slicing structure which has a total area A, and the existence, and strength (in that several wires may connect a pair of components) of wires between pairs of components are determined by the nature of the circuit and can be summed to give a total wirelength W. The cost function used by Wong and Liu is

$$C = A + \lambda W$$

where λ is a constant which can allow the significance of wiring length to be adjusted with respect to that of total area.

Clearly in taking any analogous approach to the problem of page composition, it will be appropriate to use a cost function that relates to one or more properties of an arrangement of objects on a page. The present inventors have realised that the VLSI floorplan problem is surprisingly analogous to the page composition problem, particularly when there are several objects to place on to a page and when the objects

form two or more groups. The objects take essentially the same role as the components, and the role of the wires can be taken by connections made between objects in the same group. The result of minimising function C now has the effect of minimising a combination of the total area occupied by the objects and of the proximity to each other of objects in the same group. The present inventors have found that this computationally simple approach leads to very effective results.

In a preferred arrangement "wires" of equal strength are created between each pair of objects in the same group. Minimising "wirelength" thus clearly has the effect of ensuring a close grouping of the group. However, for greater computational simplicity, or to favour a particular arrangement of objects within a group, other arrangements may be made in which objects within a group are connected only to specific other objects within that group (though each group member clearly must be connected to at least one other group member) – ring or star structures thus might be employed, for example.

For the function C to be evaluated for any given arrangement, both total area A and wirelength W must be known. Approaches to calculation of A and W in the context of page composition are discussed below.

Calculation of area A is essentially the same problem for VLSI and for page composition. Where different objects have different aspect ratios, this problem is not trivial, but a solution is indicated in Wong and Liu and will be discussed below with reference to Figures 12 and 13. In the initial part of this discussion, objects will be assumed to have constant area but a continuously variable aspect ratio.

For each individual object, a graph can be drawn of width against height, and a bounding curve can be plotted that joins all points of the desired area. This line will be a hyperbola 121, as shown in Figure 12A. Figure 12A shows the bounding curve for an unconstrained object of area 2. In practice, there will generally be constraints upon the aspect ratio, and outside upper and lower aspect ratio limits the bounding curve will be a straight line rather than a hyperbola, as increasing one dimension beyond its range will not decrease the other. Such a bounding curve 122 is shown in

Figure 12B – this shows an object with area 2 but with an aspect ratio that can vary between 0.5 and 2.0.

The region above such a bounding curve 122 represents all the possible dimensions of a rectangle that is able to contain the object concerned – any point on the hyperbolic section will be optimal, in that it will contain no wasted space. This is the position for packing of a single object, but complexity is clearly introduced when objects are combined. It is found that the bounding curve for the composite produced by any operator can be derived from the bounding curves of its two operands: if the operator is a vertical cut, then the widths must be added and the greater of the two heights used; whereas if the operator is a horizontal cut, then the heights must be added and the greater of the two widths used. The main computational cost is in adding hyperbolae – however an effective solution can be found by first order approximation of the hyperbolic section to a line, in which case only the end points of the hyperbolic section (which appears as corners) need to be calculated. Such an approximated bounding curve 123 is shown in Figure 12C.

The use of approximated bounding curves in calculating the bounding curve of a composite is shown in Figure 13. For the operation ab+, it has already been indicated that the new bounding curve is found by taking the greater of the two widths and the sum of the heights. Where a (bounding curve 131) has area 2 and an aspect ratio of between 0.5 and 2 and b (bounding curve 132) has area 8 and an aspect ration of between 0.5 and 2, it can be seen that the the composite bounding curve 133 can be found by taking the relevant corners 134,135 of b and by adding the height value from the corresponding points 136,137 in a to get the corners of the new bounding curve.

Corners from b	Corresponding value from a	New corner
(2,4)	(2,1)	(2,5)
(4,2)	(4,1)	(4,3)

Calculation of the area of an expression is therefore computationally simple. The resulting expression therefore has its own bounding curve, and as all slicing operations are essentially similar and as all approximated bounding curves have the

same form, the total area can be calculated very simply by working up from the leaves of the slicing tree to its root, calculating the approximated bounding curve at each node. The total approximated bounding curve yields the lowest area – this will simply be the point on the bounding curve for which xy is minimum.

Wirelength computation is not so mathematically complex – the only significant issue to be resolved is where wires start and finish. A logical choice is the straight line distance (though this is computationally slightly more costly than calculating a corresponding Manhattan distance) between the centres of the rectangular elements – however other choices could be made (a Manhattan distance between element centres, or a straight line distance between element corners, could be employed).

A pseudocode version of the genetic algorithm is as follows:

```
for G iterations do
        for nXC iterations do
                select two solutions
                crossover those solutions to create offspring
        endfor
        add all offspring to subpopulation
        calculate fitnesses
        select a population of n elements by fitness
        generate nXM random mutations
endfor
```

There are n elements in the population, a crossover rate C (with a value between 0 and 1 inclusive) and a mutation rate M (with a value between 0 and 1 inclusive). When a crossover is required, parents are chosen at random each time from the existing population, with the same parent being able to appear in subsequent crossover operations in the same generation, and the crossover operator used is chosen with equal probability from Change 4, Change 5 and Change 6. When selection is made from the generation containing parents and offspring, this selection is probabilistic but with higher probability of selecting elements with higher fitness, this probability difference being a user-variable selection pressure (selection pressure can be made

variable, but would generally be kept fixed in a given context). When a mutation occurs, the mutation result replaces the original expression in the population.At the end of a generation, there are still n elements present. Clearly, the algorithm can run through an indefinite number of generations – in the present case, it is chosen to run for G generations, whereas a logical alternative is for it to run until the best solution has not improved for a predetermined number of generations. Population size, crossover rate, mutation rate, selection pressure and wirelength weight (λ) can all in principle be varied by the user.

This genetic algorithm is relatively simple and many variations and enhancements are possible – one possible enhancement is that discussed in Cohoon, of dividing the whole population into several subpopulations, running a genetic algorithm such as that indicated above separately in each subpopulation, and then allowing mixing between the different subpopulations. This process forms an "epoch", with the compound algorithm being allowed to run over a number of epochs until a termination criterion of the kind indicated above is achieved. Such an approach may increase the diversity that can be achieved and reduce the likelihood of being trapped in a local minimum when a significantly better global minimum is available. The number of subpopulations and the length of an epoch are additional user variables in this arrangement.

The key datum to preserve from running the genetic algorithm is of course the best solution discovered and its total area and wirelength. For diagnostic purposes, it may also be desirable to retain the population at the start and end of a run, the genotypes of each individual in every generation in which a new best solution emerged, the number of generations until the best individual emerged and the highest fitness score in every generation.

It should be noted that although preferred embodiments involve the use of genetic algorithms to calculate an optimal arrangement of objects; aspects of the invention can be performed using other algorithmic approaches. Other iterative approaches appropriate to solving problems of this general type can be used, most obviously simulated annealing. For example, the simulated annealing approach of Wong and Liu, using only operators Change 1, Change 2 and Change 3, could be employed.

An example of an object layout (without grouping) achieved by the genetic algorithm approach is shown in Figure 14. It is notable that this layout is visually appealing as well as being spatially compact. One notable feature is that a number of columns (or rows) appear spontaneously – this would appear to be a result of the use of slicing structures, as straight cuts running along groups of objects appear to be favoured.

Layouts with grouping are illustrated in Figures 15A to 15C, which show arrangements of 20 objects divided into two groups of 10. The type of solution achieved varies considerably with variation in λ. Figure 15A shows a solution with λ=0 – unsurprisingly, this has the minimum area (259.9 in relative units) but no apparent grouping. If λ=1, grouping dominates completely and the result is as shown in Figure 15B, with complete segregation of the two groups but a much larger area of 313.7 and an inconveniently extreme aspect ratio. It is found that in this case a value of λ=0.2 leads to a very effective compromise between grouping and area, shown in Figure 15C (an area of 267.6 is achieved, less than 3% greater than for the λ=0 solution). It is found that it is significantly easier to achieve a significant compromise between minimum area and grouping when there are only two groups than when there are three or more groups. A preferred solution when three or more groups exist may be to subdivide the page into regions in each of which there will be only two groups of elements.

Where a composed page is to be displayed in an on-screen window, for example, the aspect ratio of the lowest area solution may not need to be constrained, or may be allowed to vary within wide limits. However, when a composed page is to be printed it may be desirable to constrain the aspect ratio severely. One way to achieve a result with a desired aspect ratio is to choose a different width and height pair on the approximated bounding curve describing the arrangement – instead of choosing the width and height pair that have the lowest product (and hence the lowest total area), the width and height pair can be chosen to fit into the smallest container of the desired aspect ratio (the best solution will now be that with the bounding curve that intersects a line x=ky closest to the origin). Still better results can be achieved by relaxation of the aspect ratio constraints and modifying the cost function to the form

$$C = A + \lambda W + kP$$

where P is a penalty term, and k is a user-specified weight. P is calculated as being the difference in area between the layout itself and the smallest rectangle within a specified range of aspect ratios that could contain the layout. This places the compromise between aspect ratio and area minimisation in the hands of the user, if desired.

If a solution results that does not fill a fixed-size page, then a number of choices are available to produce an end result. Clearly one solution is simply to render the result of the iterative calculation (assuming that its bounding rectangle lies within the page dimensions) with as much white space border as is necessary around it, but this will generally not be the most visually attractive solution. A more attractive solution in this circumstance will generally be to add white space according to an appropriate spacing rule, such as adding space evenly between objects– if the aspect ratio of the solution is different from the aspect ratio of the page, a different amount of white space can be added in each dimension. If the aspect ratio has been fixed, a logical solution is to scale the whole solution up or down to fit the page (though this may lead to unsatisfactory differences between, for example, the size of text on different pages). Scaling plus addition of white space is a further possibility. The aesthetic appearance of the result may be further improved by allowing the object to migrate within its rectangle of the slicing structure rather than simply being located centrally within it – this can be done, for example, to align edges or to create white space columns.

In the examples discussed above, the areas of objects have been kept constant and the aspect ratios made continuously variable within broad limits. This may not be a preferred approach for all types of object, and indeed it may be desirable to use different constraints for different types of object. For an image, it may be desirable to have a constant, or near constant, aspect ratio – however, it may be acceptable for the area of the image to vary within broader limits. One way to achieve this might be to derive the bounding curve for such a variable area object using a preferred value (say, halfway between the upper and lower bounds), and then to increase the area of those objects which have ended up with spare space in their container. Alternatively, such an object could be allowed to take any value within its range of areas, but a cost term

could be added to bias the solution in favour of picking larger images where this was consistent with low overall area and good grouping - an additional cost term proportional to "percentage of area taken up by non-images" could be chosen to have this effect. For text, it may not be appropriate to treat the object as continuously variable in aspect ratio, but for a series of different area solutions to be available at different discrete aspect ratios, according the results of word-wrapping – there is here a danger of substantially increased computational complexity, but useful approximations are available (such as to overestimate the area required by a constant according to font size).

Although page composition has been discussed here as a step independent of selection of objects to be allocated on to a given page, it is reasonable for the allocation step to be informed at least by what can be achieved in the page composition step. The number of items allocated to a page may thus be chosen to lie within the range of optimal effectiveness of the iterative process (typically between 10 and 30 items for the genetic algorithm discussed here – alternatively, if it were desirable to use fewer than 10 items, an alternative calculation process could be used), may be chosen so that items with similar dimensions or aspect ratios are preferably directed to the same page, and that the total area of the objects is less than that of the page but not significantly so.

It will be appreciated that the embodiments described above are exemplary, and that the skilled person may devise embodiments according to aspects of the invention as claimed that differ substantially from the embodiments indicated above.

CLAIMS

1. A method of composing a page, or a portion of a page, of a document, comprising:

defining a plurality of objects to be fitted on to the page and dimensional attributes of each of the objects;

establishing an arrangement of the plurality of objects such that each object lies within a separate rectangle of a slicing structure dissection of a rectangular area;

establishing a function which provides a total cost of an arrangement of the plurality of objects based on one or more properties of the arrangement; and

finding a slicing structure arrangement of the plurality of objects with a minimised total cost by means of an iterative process.

2. A method as claimed in claim 1, wherein the iterative process comprises repeated application of a genetic algorithm.

3. A method as claimed in claim 2, wherein the genetic algorithm is adapted to generate mutations of existing single arrangements and crossovers between pairs of existing arrangements.

4. A method as claimed in claim 1, wherein one of the one or more properties of the arrangement is the total area occupied by the arrangement.

5. A method as claimed in claim 1 or claim 4, wherein the plurality of objects form two or more groups, and wherein one of the one or more properties is a measure of the proximity to each other of objects which are members of the same group.

6. A method as claimed in claim 5, wherein the proximity is measured by a total distance of lines joining one group member to another group member, such that every member of a group with more than one member has at least one line joined thereto.

7. A method as claimed in claim 6, wherein each group member is joined by one and only one line to every other member of the same group.

8. A method as claimed in any of claims 1, 4 or 5, wherein one of the one or more properties is the aspect ratio of the arrangement.

9. A data carrier having thereon a computer program adapted to program a processor of a computer system to carry out the method as claimed in any of claims 1 to 8.

10. A method of composing a page, or a portion of a page, of a document, comprising:

defining a plurality of objects to be fitted on to the page and dimensional attributes of each of the objects;

establishing a function to represent a total area of an arrangement of the plurality of objects;

minimising the function to find a minimised total area arrangement; and

fitting the minimised total area arrangement to the page.

11. A method as claimed in claim 10, wherein the step of minimising the function is constrained such that the minimised total area arrangement has a similar aspect ratio to the page, and wherein the step of fitting the minimised total area

arrangement to the page comprises scaling the minimised total area arrangement.

12. A method as claimed in claim 10, wherein the step of minimising the function is constrained such that no dimension of the minimised total area arrangement is greater than a corresponding dimension of the page, and wherein the step of fitting the minimised total area arrangement to the page comprises separating adjacent objects according to a separation rule.

13. A method as claimed in claim 10, wherein the function represents other properties of the arrangement in addition to total area of the arrangement, such that minimisation of the function produces a minimised total area arrangement which is a cooptimisation of total area and said other properties.

14. A method as claimed in claim 13, wherein the plurality of objects form at least two groups, and wherein one of the other properties is a measure of the proximity to each other of objects in the same group.

15. A method as claimed in claim 13, wherein one of the other properties is the aspect ratio of the arrangement.

16. A method as claimed in claim 10 wherein minimising the function is carried out by means of an iterative process.

17. A method as claimed in claim 16, wherein the iterative process comprises repeated application of a genetic algorithm.

18. A data carrier having thereon a computer program adapted to program a processor of a computer system to carry out the method as claimed in any of claims 10 to 17.

19. A method of providing a customised document having a plurality of pages, comprising:

selecting a plurality of selected objects for inclusion in the document from a database of two-dimensional objects, and assigning each of the selected objects to one of a plurality of groups;

assigning each of the selected objects to one of the pages of the document;

arranging the objects assigned to each one of the pages in an arrangement such as to minimise a function dependent on a total area of the arrangement and on proximity to each other of objects in the same group.

20. A method as claimed in claim 19, wherein proximity to each other of objects in the same group is determined by separations between each object in the same group and at least one other object in the same group.

21. A method as claimed in claim 20, wherein proximity to each other of objects in the same group is determined by separations between each object in the same group and every other object in the same group.

22. A method as claimed in claim 19, wherein the step of arranging the objects comprises dividing the page into regions and making separate arrangements in each of the regions.

23. A method as claimed in claim 19 or claim 22, wherein an arrangement comprises objects in no more than two groups.

24. A method as claimed in claim 19, wherein the relative significance of the total area of the arrangement and of proximity to each other of objects in the same group is variable.

25. A method as claimed in claim 19, wherein said step of arranging the objects comprises establishing an arrangement of the plurality of objects such that each object lies within a separate rectangle of a slicing structure dissection of a

rectangular area and finding a slicing structure arrangement of the plurality of objects with a minimised total cost by means of an iterative process.

26. A method as claimed in claim 25, wherein the iterative process comprises repeated application of a genetic algorithm.

27. A data carrier having thereon a computer program adapted to program a processor of a computer system to carry out the method as claimed in any of claims 19 to 26.

ABSTRACT

Page Composition

A page is composed by identifying objects to be fitted on to the page and then carrying out an iterative process to minimise a cost function dependent on properties of the arrangement. Computational advantages are obtained by describing suc arrangements as slicing structures. Minimised area arrangements can be fitted to existing page dimensions by scaling or adding white space. If objects form a plurality of groups, the cost function may include a term dependent on on proximity to each other of objects in the same group.

(Figure 16)

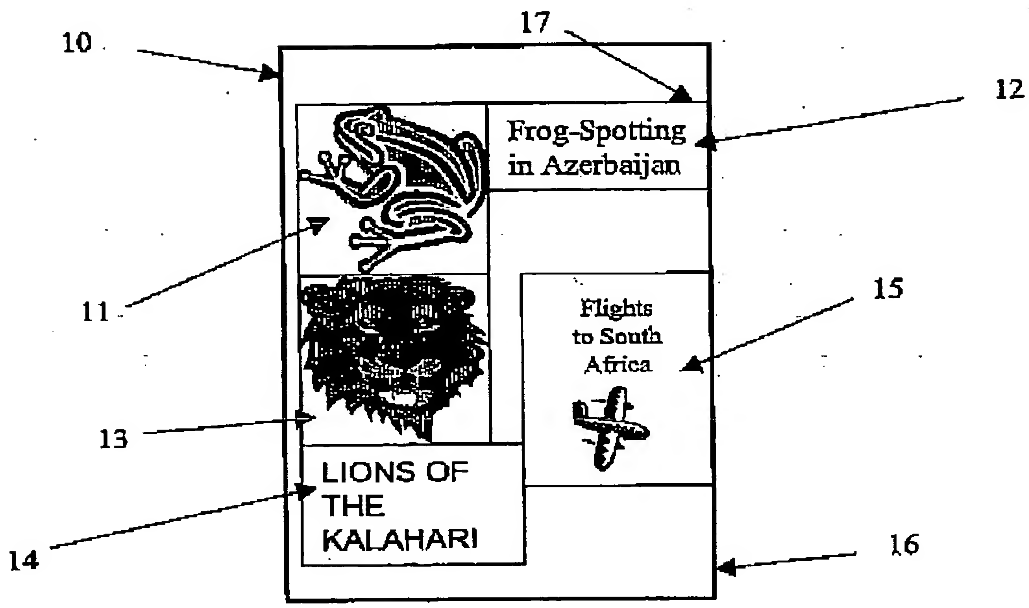

Figure 1

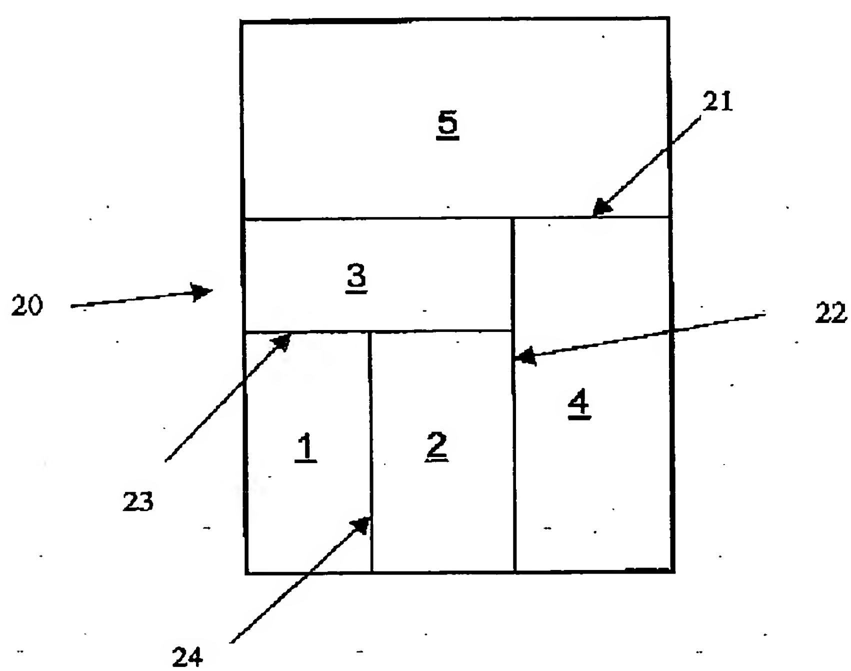

Figure 2

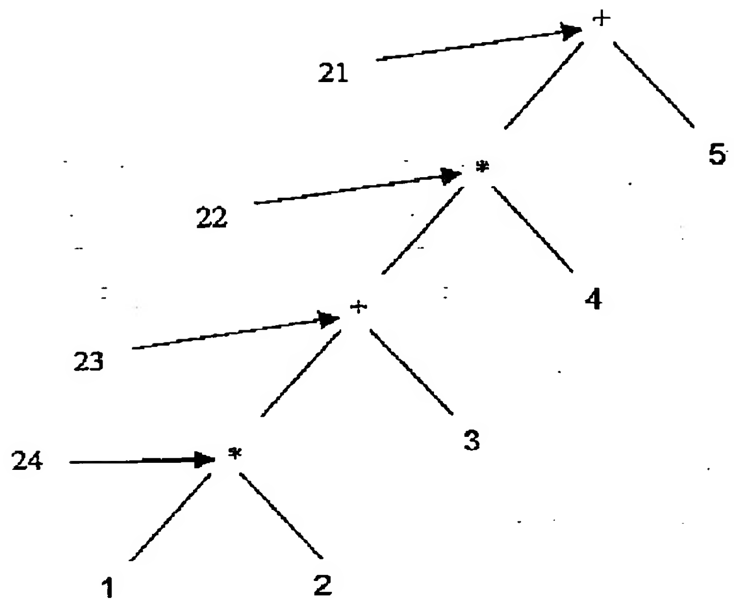

Figure 3

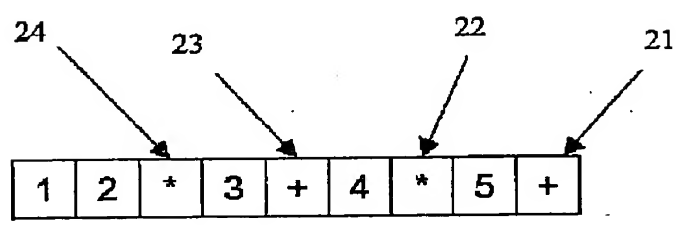

Figure 4

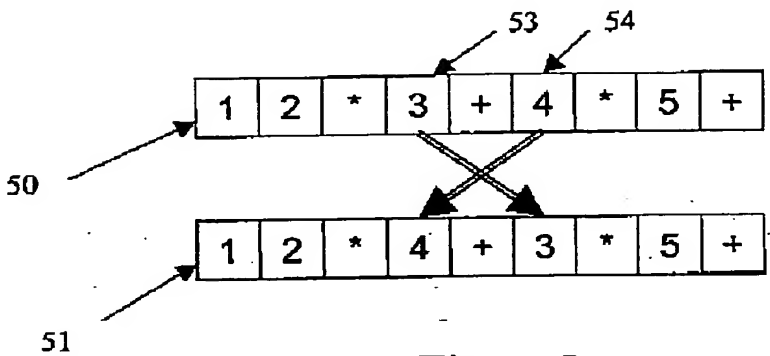

Figure 5

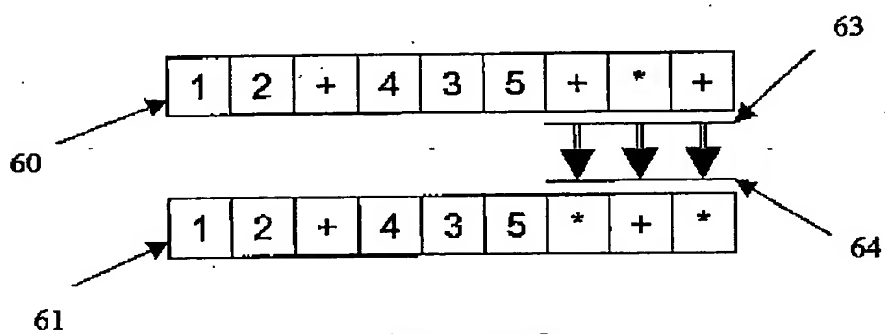

Figure 6

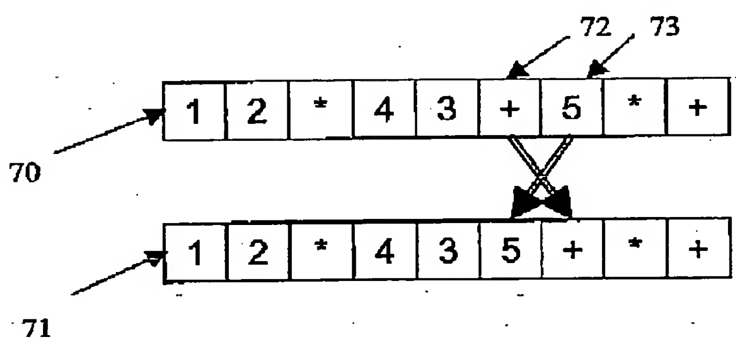

Figure 7

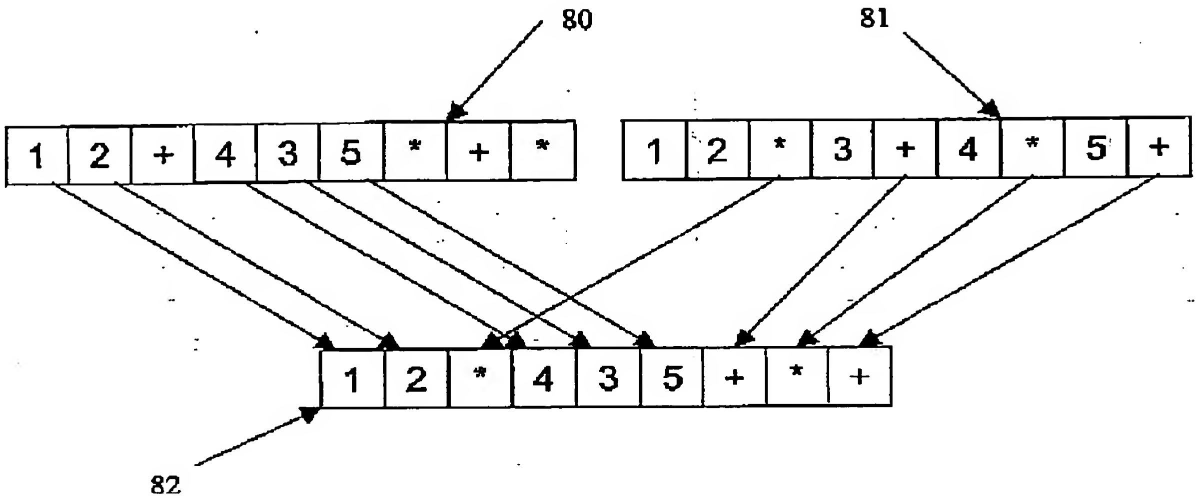

Figure 8

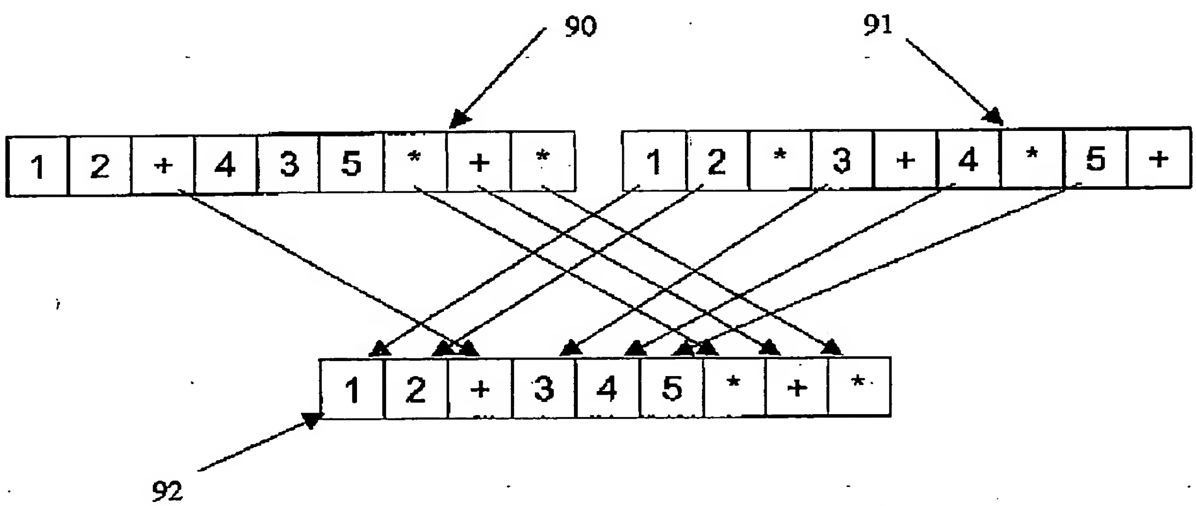

Figure 9

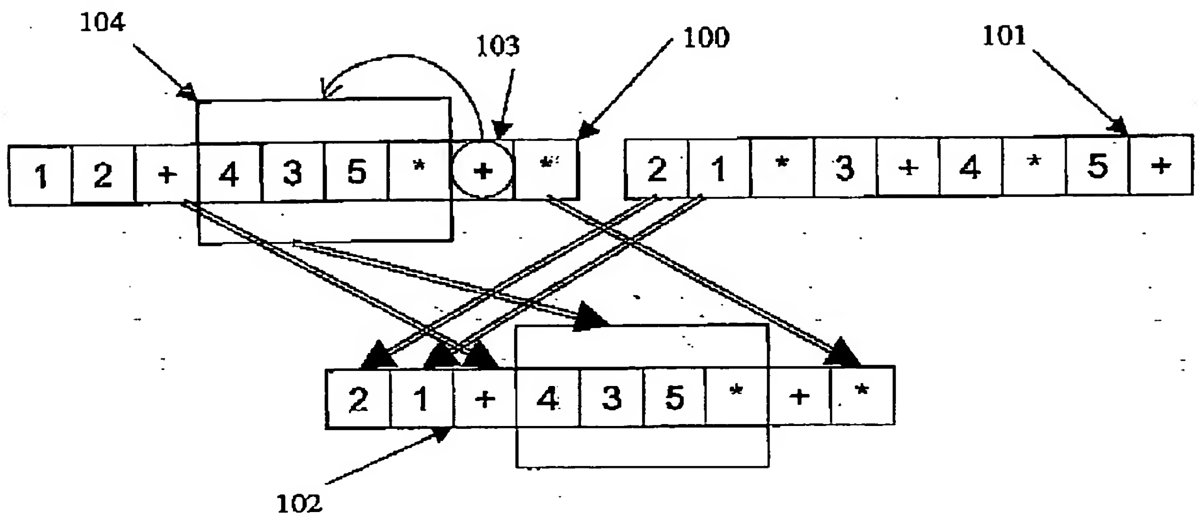

Figure 10

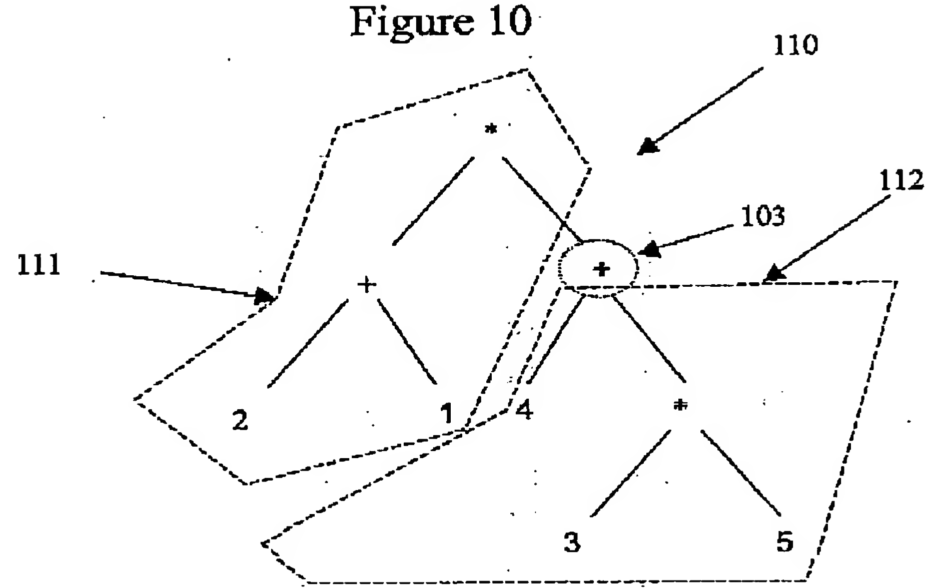

Figure 11

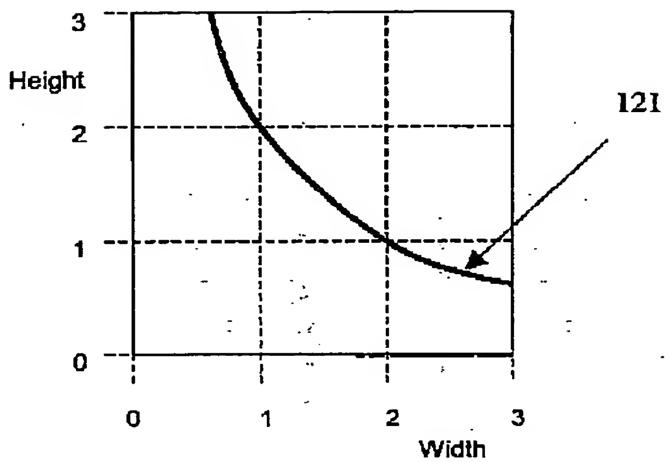

Figure 12A

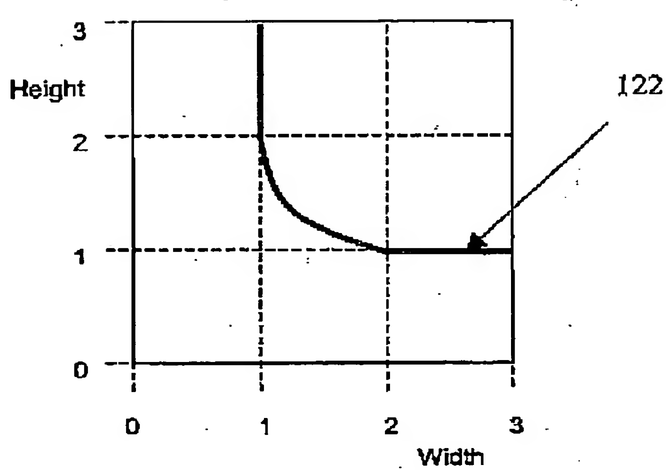

Figure 12B

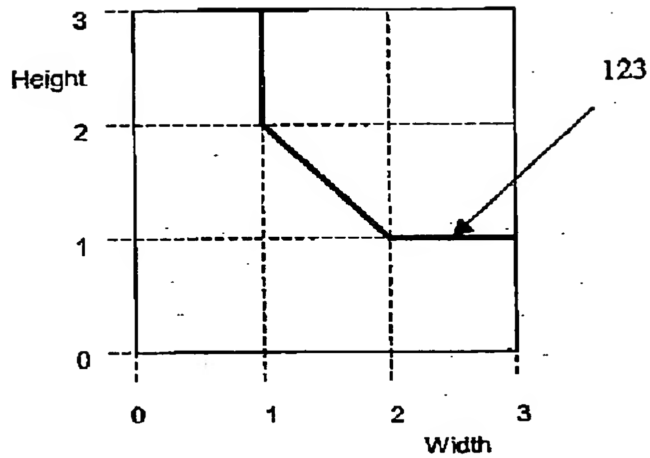

Figure 12C

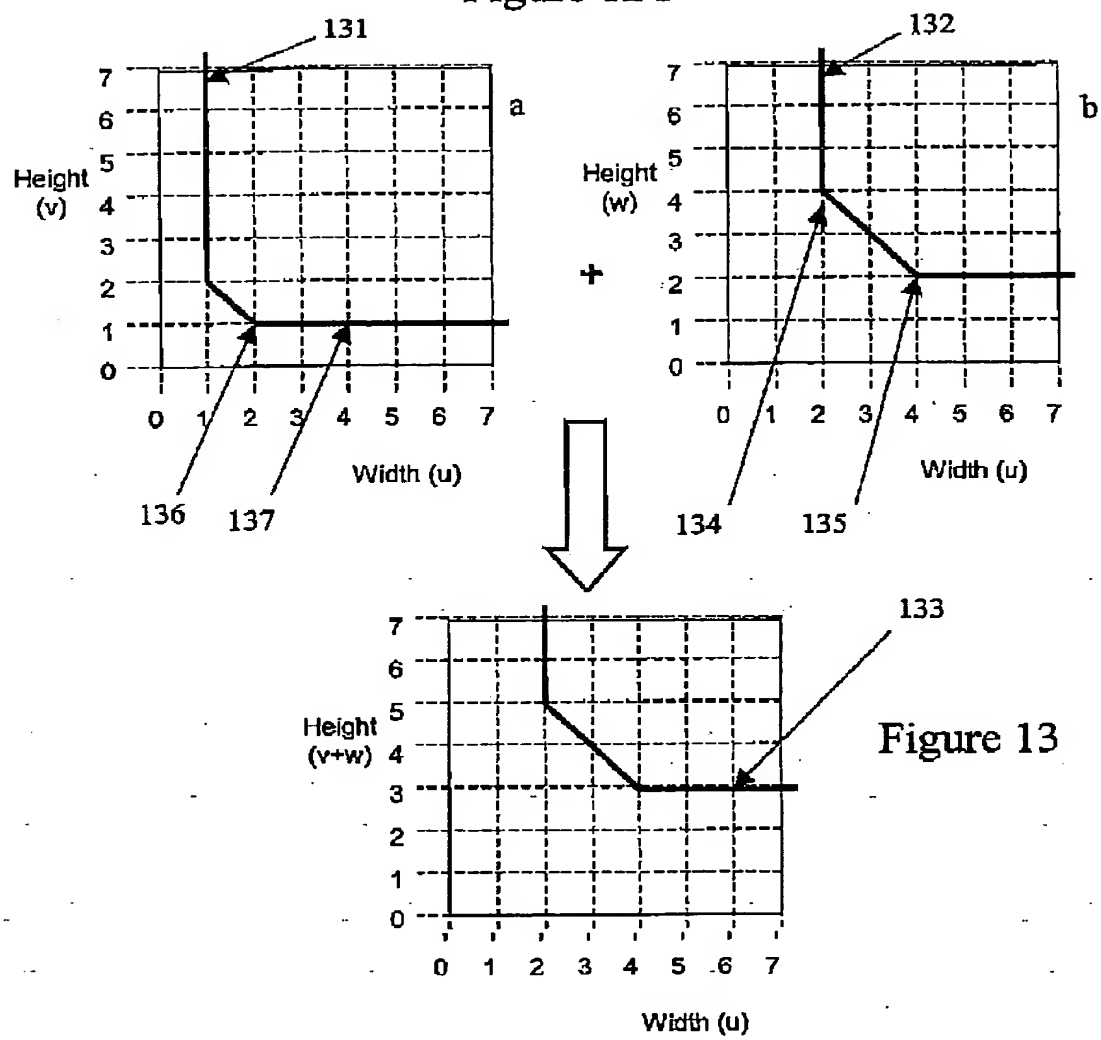

Figure 13

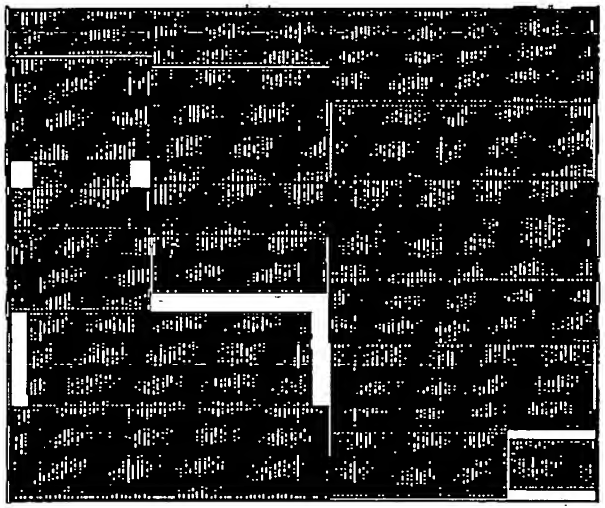

Figure 14

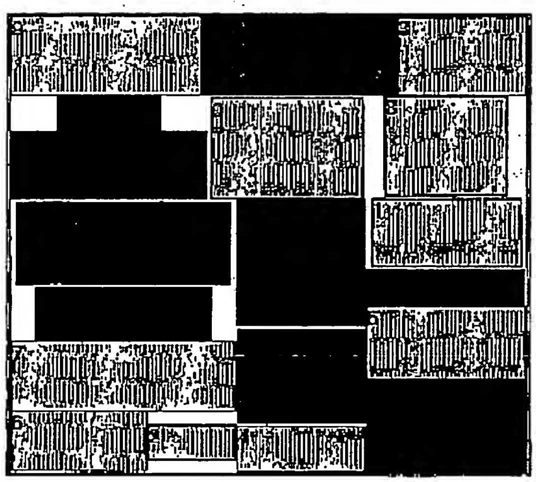

Figure 15A

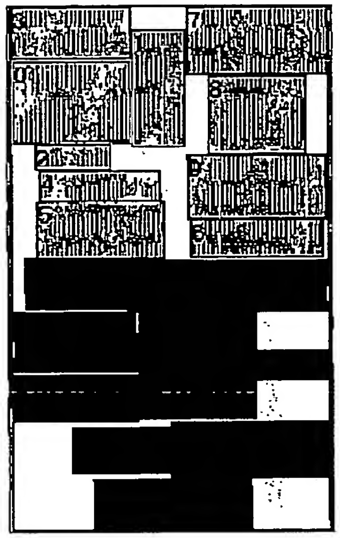

Figure 15B

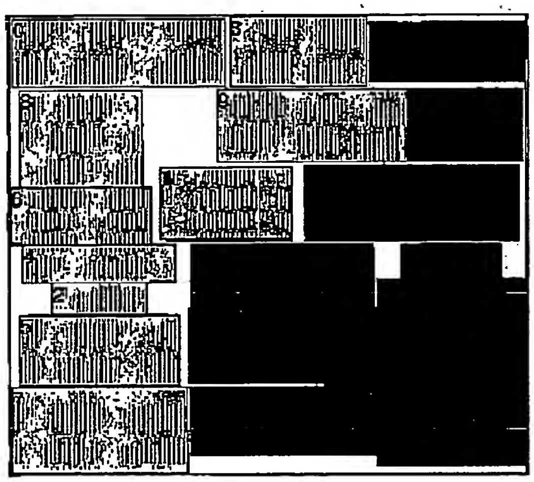

Figure 15C

SELECT CONTENT FOR DOCUMENT — 161

↓

ALLOCATE CONTENT TO GROUPS — 162

↓

ALLOCATE CONTENT TO PAGES — 163

↓

ARRANGE CONTENT ON PAGES — 164

↓

MATCH ARRANGEMENT TO PAGE DIMENSIONS — 165

Figure 16

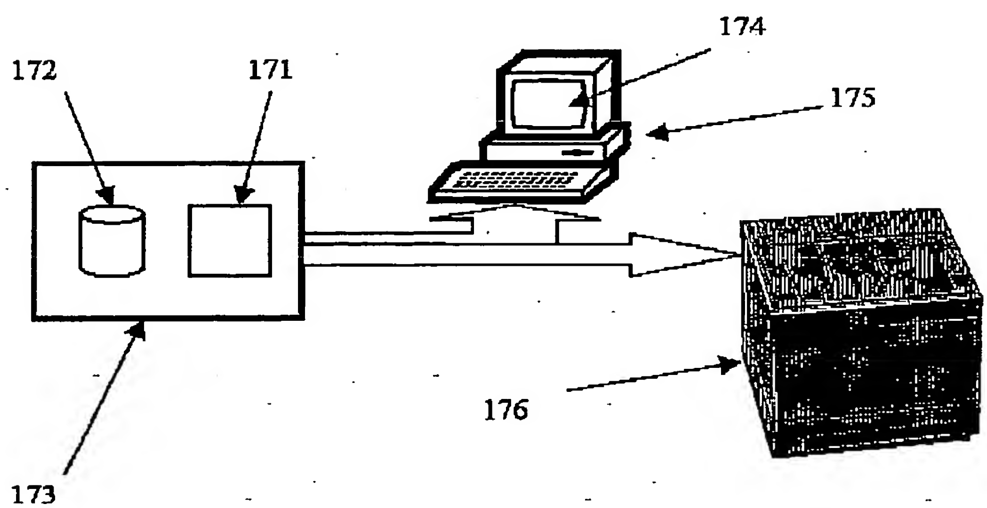

Figure 17